

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-015672

(43)Date of publication of application : 22.01.1999

(51)Int.Cl. G06F 9/445  
G06F 12/16

(21)Application number : 09-183033

(71)Applicant : BROTHER IND LTD

(22)Date of filing : 23.06.1997

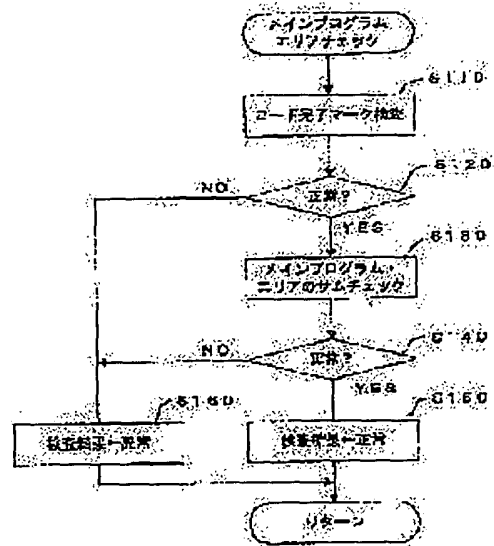
(72)Inventor : KUWABARA KAZUHIRO

## (54) DATA STORAGE STATE JUDGEMENT METHOD/DEVICE

## (57)Abstract:

**PROBLEM TO BE SOLVED:** To prevent incorrect judgment owing to accidental matching on sum checking in judgment whether data is normally written in a storage device or not.

**SOLUTION:** A specified character string is contained in the end of a main program as a loading completion mark. At the down-load the load completion mark is stored in the specified storage area of a flash memory. When the load completion mark is stored in a load completion mark storage area in the flash memory (S120: YES) and the result of sum checking on the main program area is normal (S140: YES) in judgment whether the main program is precisely stored or not, a system is judged to be normal (S150).



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the  
examiner's decision of rejection or application  
converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of  
rejection][Date of requesting appeal against examiner's decision  
of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 1 1 - 1 5 6 7 2

(43) 公開日 平成 1 1 年 ( 1 9 9 9 ) 1 月 2 2 日

(51) Int.Cl.	識別記号	庁内整理番号	F I	技術表示箇所
G06F 9/445			G06F 9/06	420 T
12/16	320		12/16	320 B

審査請求 未請求 請求項の数 5 F D (全 1 5 頁)

(21) 出願番号 特願平 9 - 1 8 3 0 3 3

(22) 出願日 平成 9 年 ( 1 9 9 7 ) 6 月 2 3 日

(71) 出願人 0 0 0 0 5 2 6 7

ブラザー工業株式会社

愛知県名古屋市瑞穂区苗代町 1 5 番 1 号

(72) 発明者 桑原 和浩

名古屋市瑞穂区苗代町 1 5 番 1 号 ブラザ  
ー工業株式会社内

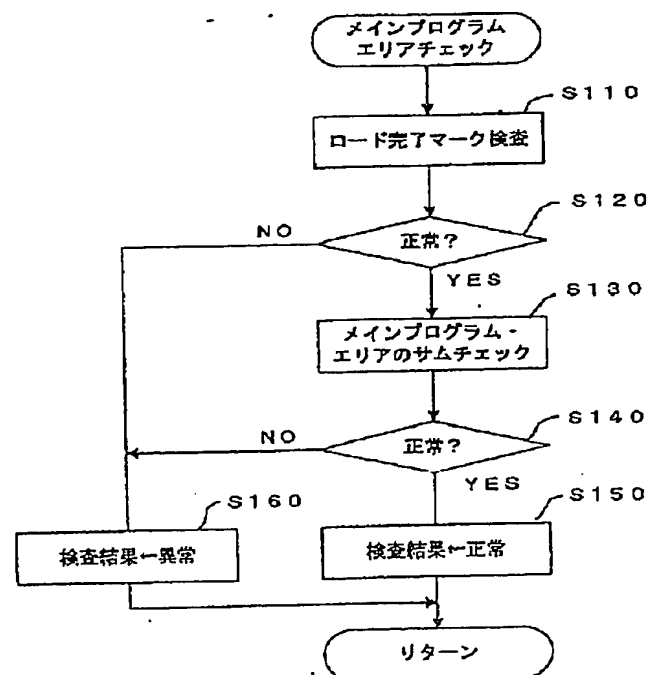
(74) 代理人 弁理士 森 泰比古

(54) 【発明の名称】 データ格納状態判定方法及び装置

(57) 【要約】

【課題】 記憶装置にデータが正常に書き込まれているか否かの判定において、サムチェックにおける偶然の一致による誤判定を防止する。

【解決手段】 メインプログラムの末尾に、ロード完了マークとして予め定めた特定の文字列を含ませておき、ダウンロードに際しては、このロード完了マークをフラッシュメモリの特定の格納領域に格納する。そして、メインプログラムが正しく格納されているか否かの判定では、フラッシュメモリ内のロード完了マーク格納領域にロード完了マークが格納され (S 1 2 0 : Y E S)、かつメインプログラムエリアに対するサムチェックの結果が正常の場合に (S 1 4 0 : Y E S)、正常と判定する (S 1 5 0)。



## 【特許請求の範囲】

【請求項 1】 書き換え可能な記憶装置に対してデータを書き込む際に、該データの本質的部分であるデータ本体、該データ本体のチェックサム及び予め定められている特定データを、それぞれについて定められた個々の格納領域に書き込み、

前記データ本体が正しく書き込まれているか否かの判定に際しては、前記チェックサムを用いて前記データ本体の格納領域に対するサムチェックを行うと共に、前記特定データの格納領域に前記特定データが格納されているか否かをも判定し、

前記サムチェックの結果が正常であっても、前記特定データの格納領域に前記特定データが格納されていないと判定された場合には、前記データ本体が前記記憶装置に正しく格納されていないものと判定するデータ格納状態判定方法。

【請求項 2】 請求項 1 記載のデータ格納状態判定方法において、前記データの書き込みに際して、前記特定データの格納領域の記憶内容を前記特定データとは異なるデータに書き換えてから前記データの書き込みを行うことを特徴とするデータ格納状態判定方法。

【請求項 3】 請求項 2 記載のデータ格納状態判定方法において、前記特定データを、前記データ本体の後に含ませておき、前記記憶装置に対して前記データ本体を書き込んだ後に、前記特定データの書き込みを行うことを特徴とするデータ格納状態判定方法。

【請求項 4】 データ本体を格納するデータ本体格納領域、該データ本体のチェックサムを格納するチェックサム格納領域、及び予め定められている特定データを格納する特定データ格納領域を備える書き換え可能な記憶手段と、

前記データ本体格納領域に対して、前記チェックサム格納領域に格納されているチェックサムを用いて前記データ本体が正常に格納されているか否かを判定するサムチェック手段と、

前記特定データ格納領域内に前記特定データが格納されているか否かを判定する特定データ判定手段と、

前記サムチェック手段により異常と判定されるか、あるいは、前記特定データ判定手段によって前記特定データ格納領域内に前記特定データが格納されていないと判定されるときは、前記データ本体が正しく格納されていないと判定する異常判定手段とを備えることを特徴とするデータ格納状態判定装置。

【請求項 5】 請求項 4 記載のデータ格納状態判定装置において、

前記記憶手段の前記各格納領域に対して、それぞれ前記データ本体、前記チェックサム及び前記特定データを書き込むデータ書込手段と、

該データ書込手段による書き込みの前に、前記特定データ格納領域の記憶内容を前記特定データとは異なるデー

タに書き換える特定データ破壊手段とを備えることを特徴とするデータ格納状態判定装置。

## 【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】本発明は、フラッシュメモリ等の書き換え可能な不揮発性記憶装置に対して、プログラムや機器の動作に必要なデータが正しく格納されているか否かを判定する方法及び装置に関するものである。

【 0 0 0 2 】

【従来の技術】従来、フラッシュメモリ等の記憶装置にプログラム等のデータをダウンロードした際に、正しくダウンロードが行われたか否かを判定する方法としてサムチェックを実行するのが一般的である。

【 0 0 0 3 】

【発明が解決しようとする課題】しかしながら、データが正しく格納されていない場合であっても、データの合計とチェックサムとが偶然一致してしまう場合がないとはいえない。このため、従来のサムチェックによる判定方法では、こうした偶然による誤判定のおそれがある。

【 0 0 0 4 】本発明は、前記問題点を解決するためになされたものであり、記憶装置にデータが正しく格納されているか否かの判定において偶然による誤判定を防止することを目的とする。

【 0 0 0 5 】

【課題を解決するための手段】かかる目的を達成するためになされた本発明のデータ格納状態判定方法は、書き換え可能な記憶装置に対してデータを書き込む際に、該データの本質的部分であるデータ本体、該データ本体のチェックサム及び予め定められている特定データを、それぞれについて定められた個々の格納領域に書き込み、前記データ本体が正しく書き込まれているか否かの判定に際しては、前記チェックサムを用いて前記データ本体の格納領域に対するサムチェックを行うと共に、前記特定データの格納領域に前記特定データが格納されているか否かをも判定し、前記サムチェックの結果が正常であっても、前記特定データの格納領域に前記特定データが格納されていないと判定された場合には、前記データ本体が前記記憶装置に正しく格納されていないものと判定するものである。

【 0 0 0 6 】この判定方法によれば、データ格納領域のデータの合計とチェックサムとが偶然一致する様な場合であっても、特定データが正しく格納されていない場合には、データ本体が正しく格納されていないものと判定する。従って、従来問題となるような偶然の一致による誤判定を防止することができる。特に、データ合計値の照合というサムチェックの手法に対し、特定データの有無の判定というサムチェックとは全く別の観点から異常を判定する構成を採用している点に一つの特徴がある。この結果、本発明によれば、種類の異なるチェック方法のそれぞれの特性を生かすことができ、単なる二重チェ

ックを越える的確な誤判定防止機能が発揮されることになる。なお、ここで、特定データは、できるだけ長いデータにしておく方がよい。

【0007】また、本発明では、上記データ格納状態判定方法において、前記データの書き込みに際して、前記特定データの格納領域の記憶内容を前記特定データとは異なるデータに書き換えてから前記データの書き込みを行うようにするとよい。この方法によれば、特定データ格納領域の記憶内容を一旦破壊することにより、データ本体の更新時には再度特定データが書き込まれない限りは、データ本体の格納状態が異常と判定されることになる。従って、データ本体を更新しようとするとき、更新途中でエラーが起こった様なときに誤って正常と判定されてしまうことがなくなる。

【0008】ここで、このデータ格納状態判定方法において、前記特定データを、前記データ本体の後に含ませておき、前記記憶装置に対して前記データ本体を書き込んだ後に、前記特定データの書き込みを行う様にするるとよい。この様に構成することで、データ本体の更新が完了しないと特定データが書き込まれないこととなり、更新途中でエラーが発生した場合の誤判定をより確実に防止することができるからである。

【0009】また、上記目的を達成するためになされた本発明のデータ格納状態判定装置は、データ本体を格納するデータ本体格納領域、該データ本体のチェックサムを格納するチェックサム格納領域、及び予め定められている特定データを格納する特定データ格納領域を備える書き換え可能な記憶手段と、前記データ本体格納領域に対して、前記チェックサム格納領域に格納されているチェックサムを用いて前記データ本体が正常に格納されているか否かを判定するサムチェック手段と、前記特定データ格納領域内に前記特定データが格納されているか否かを判定する特定データ判定手段と、前記サムチェック手段により異常と判定されるか、あるいは、前記特定データ判定手段によって前記特定データ格納領域内に前記特定データが格納されていないと判定されるときは、前記データ本体が正しく格納されていないと判定する異常判定手段とを備える。このデータ格納状態判定装置によれば、機器を作動させるためのプログラムや機器の作動に必要なデータ等が記憶手段に正しく格納されているか否かについての判定に当たり、サムチェックの他に、特定データ判定手段による判定が加わることになる。この結果、サムチェックの結果が正常であっても、特定データが正しく書き込まれていない場合は異常と判定され、先に説明した本発明方法を具体的に実施することができる。

【0010】また、このデータ格納状態判定装置において、前記記憶手段の前記各格納領域に対して、それぞれ前記データ本体、前記チェックサム及び前記特定データを書き込むデータ書込手段と、該データ書込手段による

書き込みの前に、前記特定データ格納領域の記憶内容を前記特定データとは異なるデータに書き換える特定データ破壊手段とを備える様にするるとよい。この様に構成することで、データ本体を更新する際にエラーが発生したようなとき、誤って正常と判定されることがない。

【0011】

【発明の実施の形態】次に、本発明の一実施の形態について説明する。この実施の形態は、ファクシミリ機能、プリンタ機能、複写機能等を備えたマルチ・ファンクションデバイスに対して動作プログラムをダウンロードする方法に関するものである。実施の形態として説明するマルチ・ファンクションデバイス10は、図1に示す様に、CPU11、フラッシュメモリ12、RAM13、EEPROM14、ブザーユニット15、パラレルI/Fユニット16、符号・復号器17、プリンタユニット18、モデム19、スキャナユニット20、ネットワークコントロールユニット(NCU)21及び操作パネル22を備える。

【0012】このマルチ・ファンクションデバイス10は、操作パネル22による機能選択の結果に応じて、フラッシュメモリ12に格納されたメインプログラムをCPU11が実行することにより、ファクシミリとしての機能の他に、プリンタ、複写機、スキャナ等としても機能することができるものである。

【0013】フラッシュメモリ12は、図2に示す様に、ブートローダーエリア30と、メインプログラムエリア40とから構成されている。ブートローダーエリア30には、ブートローダープログラムを格納するブートローダープログラム格納領域31の他に、ブートローダープログラムの作成時期を表す「作成時期データ」を格納するためのブートローダー作成時期格納領域32と、ブートローダーエリアについての「チェックサムデータ」を格納するためのブートローダーチェックサム格納領域33が設定されている。また、メインプログラムエリア40には、メインプログラムを格納するメインプログラム格納領域41の他に、本実施の形態の特徴である「モデル・国識別用パスワード」を格納するパスワード格納領域42と、同じく本実施の形態の特徴であるメインプログラムのロード完了を示す「ロード完了マーク」を格納する完了マーク格納領域43と、メインプログラムの作成時期を表す「作成時期データ」を格納するためのメインプログラム作成時期格納領域44と、メインプログラムエリア40についての「チェックサムデータ」を格納するためのメインプログラムチェックサム格納領域45とが設定されている。

【0014】EEPROM14は、ユーザーのパネル操作によって設定可能なワンタッチダイヤル等の情報を格納するためのものである。また、ブザーユニット15は、プログラムのダウンロード完了時やエラー発生時に報知音をスピーカ23から出力するためのものである。

10

20

30

40

50

パラレル I / F ユニット 1 6 は、パラレルポート 2 4 を介してパーソナルコンピュータとの間でデータのやり取りを行うために設けられている。メインプログラムは、後述する様に、このパラレル I / F ユニット 1 6 を介してパーソナルコンピュータからダウンロードされる。また、本実施の形態のマルチ・ファンクションデバイス 1 0 は、このパラレル I / F ユニット 1 6 を介してパーソナルコンピュータから印字データを受信した場合はプリンタとして機能する。

【 0 0 1 5 】 符号・復号器 1 7 は、ファクシミリとして機能する際に、スキャナユニット 2 0 にて原稿から読み取ったデータをファクシミリ送信用のデータに符号化する処理と、外部から受信したファクシミリデータをプリンタユニット 1 8 で印字出力可能な印字データに復号化する処理とを行うためのものである。この符号・復号器 1 7 によって符号化されたファクシミリデータは、モデム 1 9 及び N C U 2 1 を介して、一般電話回線 2 5 で送信される。また、モデム 1 9 及び N C U 2 1 を介して、一般電話回線 2 5 から受信したファクシミリデータは、符号・復号器 1 7 で復号化され、プリンタユニット 1 8 に送られて記録用紙に印字される。

【 0 0 1 6 】 操作パネル 2 2 には、電話番号を入力するためのテンキーの他、スタートキー、ストップキー、機能選択キー等が設けられている。この操作パネル 2 2 によって機能を選択することにより、本装置 1 0 を、ファクシミリ、プリンタ、複写機等として機能させることができる様になっている。また、テンキー等による入力値や各種メッセージを表示するための液晶ディスプレイもこの操作パネル 2 2 の中に備えられている。

【 0 0 1 7 】 次に、このように構成される本装置 1 0 の製造方法を含め、メインプログラムをダウンロードする手順等について説明する。

【 0 0 1 8 】 まず、本装置 1 0 に備えられるフラッシュメモリ 1 2 は、図 3 に示す様に、ワークステーション W S にフラッシュメモリープログラマー 5 0 を接続し、ブートローダープログラムを格納したマスタチップ 5 1 を製造する。続いて、このマスタチップ 5 1 をフラッシュメモリープログラマー 5 2 にセットして、コピーチップ 5 3, 5 3, ... を製造する。そして、このコピーチップ 5 3 を、本装置 1 0 用の基板 5 4 に実装し、これを本装置 1 0 に組み付ける。

【 0 0 1 9 】 次に、ワークステーション W S で作成したメインプログラムを、データベース 5 6 にアップロードする。そして、このデータベース 5 6 に対してパーソナルコンピュータ P C を接続し、メインプログラムをパーソナルコンピュータ P C にダウンロードする。こうしてパーソナルコンピュータ P C にメインプログラムをダウンロードしたら、この P C と本装置 1 0 とをプリンタケーブル 5 7 で接続し、装置 1 0 の電源を投入する。すると、後述する様に、ブートローダープログラムが起動さ

れ、メインプログラムの初期ロードが実行される。

【 0 0 2 0 】 また、こうして製造された装置 1 0 に対して、ユーザーに供給した後のプログラムのアップデートに当たっても、図 4 に示す様に、パーソナルコンピュータ P C を使用することができる。メーカーは、アップデート用プログラムの提供時期、方法等について、自己のホームページに情報を表示する。この表示を見たユーザーは、インターネット経由で、アップデート用プログラムを自己のパーソナルコンピュータ P C 内にダウンロードする。そして、このパーソナルコンピュータ P C と装置 1 0 とをプリンタケーブル 5 7 で接続し、ダウンロードを実行する。あるいは、他の方法として、メーカーがフロッピーディスクあるいは C D R O M 等の可搬式記憶媒体 5 8 にアップデート用プログラムを格納し、これを各ユーザーに供給する。各ユーザーは、この記憶媒体 5 8 を自己のパーソナルコンピュータ P C にセットし、後は、インターネット経由の場合と同様に装置 1 0 に対してプリンタケーブル 5 7 を介してアップデート用プログラムのダウンロードを実行する。

【 0 0 2 1 】 なお、このマルチファンクション・デバイス 1 0 は、複数のモデルを有し、メインプログラムは各モデル毎に異なるものである。また、各モデルについて、さらに、仕向国別にメインプログラムが異なるものとして作成されている。しかしながら、これらメインプログラムは、いずれも、上述のブートローダープログラムを使ってダウンロードすることが可能である。この結果、本実施の形態では、製品に組み付ける基板 5 4 に実装されるコピーチップ 5 3 は全て共通のものを使用する。

【 0 0 2 2 】 ここで、ワークステーション W S で作成されるメインプログラムは、図 5 に示す様に、データの先頭から順番に、P J L コマンド、固定文字列、モデル・国識別用パスワード、開始レコード、データレコード、最終レコードによって構成されている。

【 0 0 2 3 】 P J L コマンドは、このコマンドに続くデータが印字データではなくプログラムであることを C P U 1 1 に認識させるためのコマンドである。続く固定文字列は、本装置 1 0 用のプログラムであることを識別させるためのデータである。

【 0 0 2 4 】 さらに、その後に続くモデル・国識別用パスワードは、データレコードとして格納されているプログラムが、本装置 1 0 におけるどのモデルの、どの国用のものであるかを識別するためのパスワードである。開始レコードと最終レコードは、データレコード領域の開始と終了とを識別するためのものである。

【 0 0 2 5 】 データレコードは、属性フィールド、アドレスフィールド、データフィールド、チェックサムフィールドから構成されている。属性フィールドには、データレコードであることを示すデータとデータフィールド内のデータ数を判別するためのデータとが記憶される。

アドレスフィールドには、データフィールド内のデータのフラッシュメモリ上での格納アドレスに関するデータが記憶されている。チェックサムフィールドには、属性フィールド、アドレスフィールド及びデータフィールドのデータを加算した合計の値を示すデータが記憶されている。

【 0 0 2 6 】そして、データフィールドには、プログラム、モデル・国識別用パスワード、メインプログラムロード完了マーク、メインプログラム作成時期データ、メインプログラムエリアのチェックサムデータといったものが記憶されている。これらの並び方は、開始レコード側から最終レコード側にかけて、プログラム、モデル・国識別用パスワード、メインプログラムロード完了マーク、メインプログラム作成時期データ、メインプログラムエリアのチェックサムデータの順となっている。

【 0 0 2 7 】次に、メインプログラムの初期ロードを含む、本装置 1 0 の制御処理の内容について図 6 ～図 1 0 のフローチャートを用いて説明する。

【 0 0 2 8 】本装置 1 0 に電源が投入されると、図 6 に示す処理が開始され、まず、I/Oポートの初期化が実行される ( S 1 0 ) 。次に、フラッシュメモリ 1 2 のブートローダーエリア 3 0 についてサムチェックを行う ( S 2 0 ) 。そして、ブートローダーエリア 3 0 の内、チェックサムデータ格納領域 3 3 を除くエリアの全データを加算した合計とチェックサムとが一致している場合に正常と判定し ( S 3 0 : Y E S ) 、続いて R A M 1 3 がデバイスとして正常に動作し得るか否かを検査する ( S 4 0 ) 。この R A M 1 3 の検査も正常であるとき ( S 5 0 : Y E S ) 、後述するメインプログラムエリアチェック処理が実行される ( S 1 0 0 ) 。

【 0 0 2 9 】そして、メインプログラムエリアチェック処理の結果が正常であるとき ( S 6 0 : Y E S ) 、後述のメインプログラム実行ルーチンが起動される。一方、メインプログラムエリア 4 0 に対するチェックの結果が異常であるときは ( S 6 0 : N O ) 、後述のダウンロード処理へと移行する ( S 2 0 0 ) 。なお、ブートローダーエリア 3 0 のサムチェック結果が異常であるとき ( S 3 0 : N O ) 、及び R A M 1 3 の検査結果が異常であるときは ( S 5 0 : N O ) 、装置異常処理ルーチンが起動される。この装置異常処理ルーチンは、デバイスの交換等を指示するもので、ブザーの鳴動やエラーメッセージのディスプレイ表示あるいは印字出力等、所定の処理が実行される。

【 0 0 3 0 】メインプログラムエリアに対するチェック処理は、図 7 に示す手順で実行される。まず、フラッシュメモリ 1 2 内のロード完了マーク格納領域 4 3 にロード完了マークが格納されているか否かを検査する ( S 1 1 0 ) 。この検査の結果、ロード完了マークが正常に格納されているときは ( S 1 2 0 : Y E S ) 、メインプログラムエリア 4 0 に対するサムチェックを実行する ( S

1 3 0 ) 。そして、メインプログラムエリア 4 0 の内、チェックサムデータ格納領域 4 5 を除くエリアの全データを加算した合計とチェックサム格納領域 4 5 に格納されているチェックサムとが一致しているときは ( S 1 4 0 : Y E S ) 、検査結果を正常に設定して本ルーチンを抜ける ( S 1 5 0 ) 。一方、ロード完了マークが正しく格納されていないと判定されたとき ( S 1 2 0 : N O ) 、及びメインプログラムエリア 4 0 に対するサムチェックの結果が異常であると判定されたときは ( S 1 4 0 : N O ) 、検査結果を異常に設定して本ルーチンを抜ける ( S 1 6 0 ) 。上述の S 6 0 の処理では、こうして S 1 5 0 , S 1 6 0 で設定された検査結果に基づき、メインプログラム実行ルーチン及びダウンロード処理のいずれのルーチンを起動すべきかが判定されることになる。

【 0 0 3 1 】ダウンロード処理ルーチンが起動された場合、図 8 に示す様に、まず、初期ロードとしてダウンロードを実行すべき状態であるのか、アップデートとしてダウンロードを実行すべき状態であるのかが判定される ( S 2 1 0 ) 。 S 6 0 で N O と判定されたときは、初期ロードに失敗している状態であることから、この S 2 1 0 の判定の後、 S 2 2 0 へと処理が進む。

【 0 0 3 2 】 S 2 1 0 で初期ロードと判定されたときは、まず、初期ロードであることを特定するために動作モードを「 0 」に設定すると共に、フラッシュメモリ 1 2 のセクタがいずれも消去されていないことを判別させるための消去フラグを「 0 」に設定する ( S 2 2 0 ) 。その後、データ待ちカウンタを「 0 」に設定すると共に ( S 2 3 0 ) 、パスワード照合に関する条件を「照合しない」に設定する ( S 2 4 0 ) 。

【 0 0 3 3 】そして、まず、フラッシュメモリ 1 2 に対して正常に読み書き動作を実行できるか否かのデバイスチェックを実行する ( S 2 5 0 ) 。このデバイスチェックの結果が異常であるときは ( S 2 6 0 : N O ) 、装置異常処理ルーチンを起動する。一方、デバイスチェックの結果が正常であるときは ( S 2 6 0 : Y E S ) 、後述のダウンロード実行処理ルーチンを起動する ( S 4 0 0 ) 。そして、ダウンロード実行処理ルーチンが正常に終了した場合は ( S 2 7 0 : Y E S ) 、先に説明したメインプログラムエリア 4 0 に対するチェック処理を実行し ( S 1 0 0 ) 、その結果が正常であるときは ( S 2 8 0 : Y E S ) 、電源投入・リセット処理ルーチンへ戻って、メインプログラム実行ルーチンを起動する。

【 0 0 3 4 】一方、ダウンロード実行処理ルーチンが正常終了しなかったとき ( S 2 7 0 : N O ) 、及びメインプログラムエリア 4 0 に対するチェック処理の結果が異常であるときは ( S 2 8 0 : N O ) 、ダウンロード異常を報知するためにブザー 2 3 を鳴動させる ( S 2 9 0 ) 。そして、設定されている動作モードに応じて S 2 2 0 または S 3 1 0 へとジャンプする。

10

20

30

40

50

【0035】S310～S330の処理は、初期ロードではなく、アップデートとしてメインプログラムをダウンロードする場合に実行されるステップである。アップデートとしてメインプログラムのダウンロードを実行するときは、初期ロードと区別するために動作モードを「1」に設定すると共に、フラッシュメモリ12のセクタがいずれも消去されていないことを判別させるための消去フラグを「0」に設定する（S310）。その後、データ待ちカウンタを「0」に設定すると共に（S320）、パスワード照合に関する条件を「照合する」に設定する（S330）。

【0036】次に、ダウンロード実行ルーチンについて図9のフローチャートに従って説明する。このルーチンでは、まず、パラレルポート24からプログラムデータを受信可能な状態に準備する（S410）。続いて、ダウンロード開始を報知するためにブザーを鳴動させ（S420）、固定文字列を受信するのを待つ（S430）。固定文字列を受信したら（S430：YES）、パスワード照合を実行するモードが設定されているか否かを判定する（S440）。前述の様に、パスワード照合のモードとしては、初期ロードの場合には「照合しない」が設定され（S240）、アップデートの場合には「照合する」が設定されている（S330）。

【0037】従って、アップデートの場合に限ってモデル・国識別用パスワードの受信が実行され、メインプログラムエリア40の該当領域42に格納されているパスワードと照合される（S450）。そして、照合の結果、パスワードが一致していないときは（S460：NO）、アップデート用として選択されたメインプログラムの種類が間違っていることを報知し（S470）、受信したデータを読み捨てる処理を実行する（S480）。

【0038】一方、照合の結果、パスワードが一致するときは（S460：YES）、ロード完了マークの格納されている領域43に、ロード完了マークとは異なる内容の文字列を書き込むことにより、当該領域43のデータを破壊する処理を実行する（S490）。そして、開始レコードを受信するのを待った後（S500）、1セクタ分のデータを受信し（S510）、このデータを書き込むべき1セクタの領域内のデータを消去し（S520）、消去フラグを「1」に設定すると共にブザーを鳴動させる（S530）。そして、S510で受信した1セクタ分のデータを先ほど消去を行ったセクタに書き込み（S540）、書き込み完了ブザーを鳴動させる（S550）。そして、最終レコードを受信するまでは（S560：NO）、上記S510以下の処理を繰り返し実行する。

【0039】なお、フローチャートでは処理手順が複雑になって理解し難くならないように記載していないが、このS510～S560の間の処理において、パラレル

ポート24からデータを受信するに当たってデータ待ちタイマをセットしておき、データ待ちタイマがタイムアップしてもデータを受信できないときは、データ待ちカウンタをインクリメントし、このデータ待ちカウンタの値が所定値になったとき、未だいずれのセクタも消去していない場合は、電源投入・リセットルーチンへ戻ることになっている。セクタを未だ消去していない場合は、ダウンロードを全く開始していないのと同じであるので最初から実行し直せばよいからである。

10 【0040】一方、既にいずれかのセクタを消去している場合には、ダウンロード実行処理を異常終了として、S270に進む様に構成してある。これはダウンロード途中での異常終了であることから、特に、アップデートの場合にはメインプログラムの一部が破壊された状態にあり、電源投入直後の状態に復帰させると装置10が正常に動作できなくなってしまうおそれがあることから、速やかにメインプログラムをダウンロードすべく、ダウンロード処理ルーチンを再度スタートさせるためである。

20 【0041】こうしてS510～S560の処理を繰り返し、最終レコードを受信したときは（S560：YES）、メインプログラムエリア40の中の未消去のセクタを全て消去し（S570）、ダウンロード完了を意味するブザーを鳴動させて本ルーチンを抜ける（S580）。このときは正常終了となる。よって、S270はYESと判定され、メインプログラムエリアチェック処理を実行し（S100）、データが正しくダウンロードされたか否かを判定する処理に進む。

30 【0042】次に、メインプログラム実行ルーチンについて、図10のフローチャートに基づいて説明する。このルーチンでは、外部からPJLコマンドを受信したか否かを判定し（S610）、PJLコマンドを受信した場合に（S610：YES）、前述のダウンロード処理ルーチンを起動する（S200）。これが、アップデートの状態であり、ダウンロード処理ルーチンのS210でアップデートと判定され、S310以下の処理が実行されることになる。

40 【0043】一方、印字データを受信したときは（S620：YES）、受信した印字データに基づいてプリンタユニット18を制御して印字処理を実行する（S700）。また、NCU21及びモデム19を介してファクシミリデータを受信した場合は（S630：YES）、符号・復号器17に当該ファクシミリデータを入力して復号化を行った後にプリンタユニット18を制御して受信した情報を記録紙に印字して出力するファクシミリ受信処理を実行する（S800）。また、操作パネル22においてファクシミリ機能が選択されているときは（S640：YES）、スキャナユニット20に投入された原稿からデータを読み取り、これを符号・復号器17に入力して符号化し、さらに、モデム19及びNCU21

を制御してファクシミリデータを送信する処理を実行する ( S 9 0 0 ) 。また、その他の機能、例えば、複写機能等が選択されているときは ( S 6 5 0 ) 、当該選択された機能に従って、スキャナユニット 2 0 やプリンタユニット 1 8 等を制御し、選択された機能に従った処理を実行する ( S 1 0 0 0 ) 。以上の処理は、電源が O F F にされるリセットが行われるまで続行される ( S 6 6 0 ) 。

【 0 0 4 4 】 以上説明した様に、本実施の形態では、メインプログラムの末尾に、ロード完了マークとして予め10 定めた特定の文字列を含ませておき、ダウンロードに当たっては、このロード完了マークをフラッシュメモリ 1 2 の特定の格納領域に格納する様にしている。そして、メインプログラムが正しく格納されているか否かの判定に当たっては、サムチェックだけでなく、このロード完了マークの書き込みが正常か否かについてもチェックする様にしている。ここで、サムチェックの場合、単純にデータの数を合計した値とチェックサムとを比較するだけであるため、偶然の一致による誤判定の危険性がある。しかし、本実施の形態によれば、さらにロード完了20 マークとしての文字列が格納されているか否かをもチェックすることで、こうした偶然の一致による誤判定を的確に防止することができるようになる。

【 0 0 4 5 】 また、本実施の形態では、アップデートの際にプログラムをダウンロードする前に、このロード完了マークを一旦破壊する構成を採用している。しかも、ロード完了マークをプログラムデータの末尾に配置することで、プログラムが正常にダウンロードされないでデータの受信が中断されてしまった様なときは、正しいロ30ード完了マークが格納されていない状態になる。この結果、本実施の形態によれば、アップデートの際に、ダウンロード途中でエラーが発生した様なときにも誤判定を防止することができる。

【 0 0 4 6 】 さらに、本実施の形態によれば、メインプログラムの末尾にモデル・国識別用パスワードをデータとして含ませておき、初期ロードの際に、このモデル・国識別用パスワードをフラッシュメモリ 1 2 の特定の格納領域に書き込む様にしている。そして、プログラムデータの前に、このモデル・国識別用パスワードを配置したデータでアップデートを行う様に構成し、アップデ40ート時には最初にパスワードの照合を行う様にしている。この結果、アップデートの際に、異なるモデル・国の仕様に合わせたメインプログラムを間違えてダウンロードしてしまうといった不具合を防止することができる。また、初期ロードの際には、このパスワード照合を行わないモードを設定することで、モデル・国の別に関係なく製造したフラッシュメモリ 1 2 を用いることができる。この結果、フラッシュメモリ 1 2 自体は、各製品に共通のものとして採用することができ、しかも、アップデートにおける誤りを的確に防止し得るという優れた効果が50

発揮される。

【 0 0 4 7 】 以上、本発明の一実施の形態について説明したが、本発明はこの実施の形態に限られるものではなく、その要旨を逸脱しない範囲内においてさらに種々の形態を採用することができることはもちろんである。

【 0 0 4 8 】 例えば、上述の実施の形態ではロード完了マークをプログラムデータの後ろに配置し、メインプログラムエリア 4 0 の最後の部分に確保した格納領域に格納する様にしたが、ロード完了マークをプログラムデータの前に配置する構成を採用し、その格納領域をメインプログラムエリアの先頭部分に確保する様にしてもよい。但し、アップデート時のダウンロード途中のエラーに対して誤判定しない様にする上では、実施の形態に示した様に、ロード完了マークをプログラムデータの後ろに配置する構成を採用する方が優れている。また、実施の形態ではロード完了マークとして「 P r o g r a m O K 」として、それ自体ロード完了を意味する文字を連ねたものを用いたが、それ自体が特定の意味を表す文字列でなくても構わないし、もっと短い文字列、例えば 1 文字であっても構わない。加えて、プログラム以外の、機器の動作に必要なデータをダウンロードする場合の誤判定防止方法としても本発明を適用することができる。

【 0 0 4 9 】

【発明の効果】 以上説明した様に、本発明によれば、サムチェックによる異常判定に加えて、データ合計値の照合というサムチェックにおける手法とは全く別の手法として特定データの有無をも判定する構成を採用することで、サムチェックでは防止できなかった偶然の一致による誤判定を防止することができる。

【図面の簡単な説明】

【図 1】 実施の形態としてのマルチファンクション・デバイスの構成を示すブロック図。

【図 2】 実施の形態におけるフラッシュメモリの構造を示す模式図。

【図 3】 実施の形態における製品製造時のプログラムのダウンロードの手法を示す模式図。

【図 4】 実施の形態におけるアップデート時のプログラムのダウンロードの手法を示す模式図。

【図 5】 実施の形態におけるメインプログラムのデータ構造を示す模式図。

【図 6】 実施の形態において実行する制御処理の手順を示すフローチャート。

【図 7】 実施の形態において実行する制御処理の手順を示すフローチャート。

【図 8】 実施の形態において実行する制御処理の手順を示すフローチャート。

【図 9】 実施の形態において実行する制御処理の手順を示すフローチャート。

【図 1 0】 実施の形態において実行する制御処理の手順を示すフローチャート。

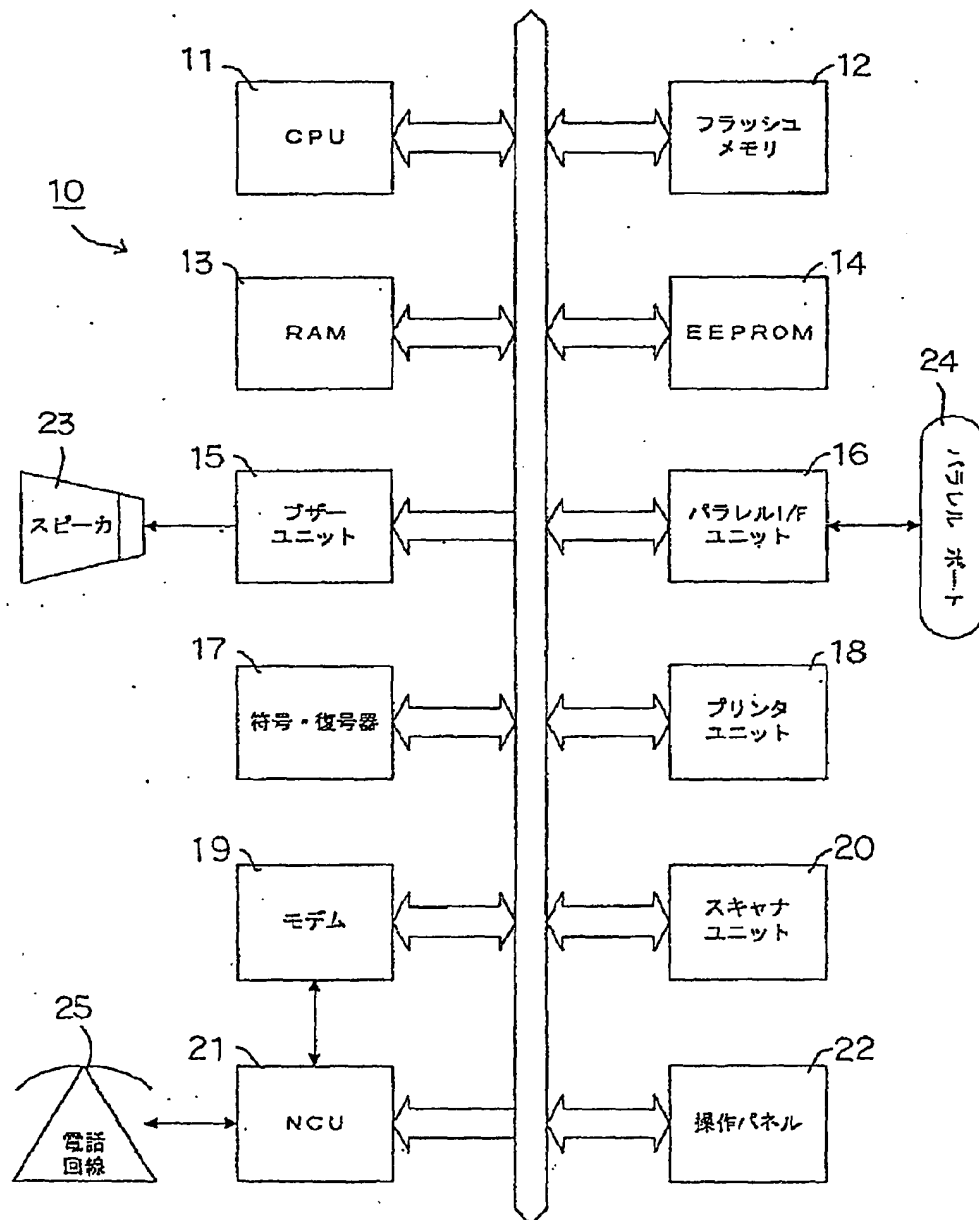


## 【符号の説明】

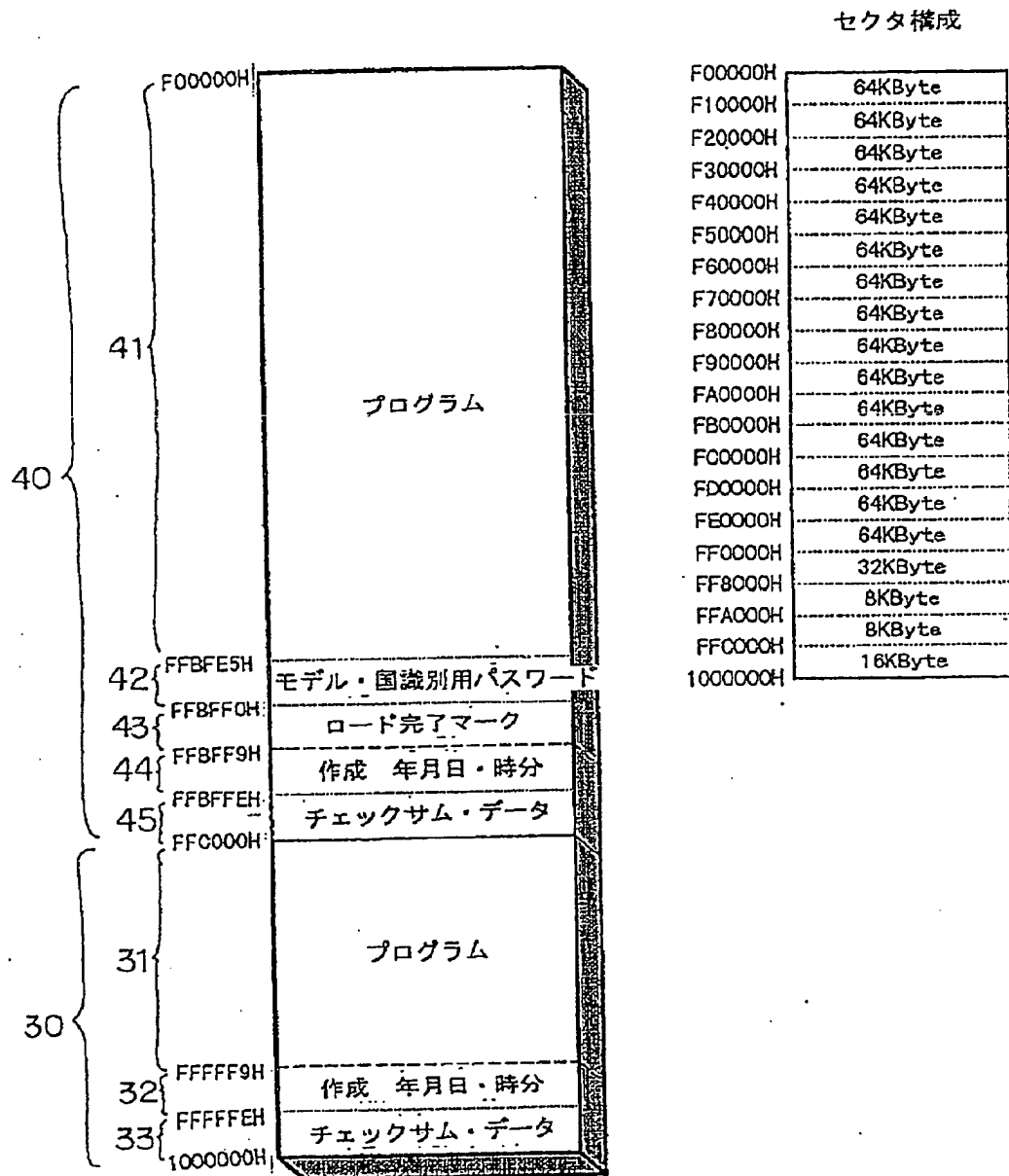
10・・・マルチ・ファンクションデバイス、11・・・CPU、12・・・フラッシュメモリ、13・・・RAM、14・・・EEPROM、15・・・ブザーユニット、16・・・パラレルI/Fユニット、17・・・符号・復号器、18・・・プリンタユニット、19・・・モデム、20・・・スキャナユニット、21・・・NCU、22・・・操作パネル、23・・・スピーカ、24・・・パラレルポート、25・・・一般電話回線、30・・・ブートローダーエリア、31・・・ブートローダープログラム格納領域、32・・・ブートローダー作

成時期格納領域、33・・・ブートローダーチェックサム格納領域、40・・・メインプログラムエリア、41・・・メインプログラム格納領域、42・・・パスワード格納領域、43・・・ロード完了マーク格納領域、44・・・メインプログラム作成時期格納領域、45・・・メインプログラムチェックサム格納領域、51・・・マスタチップ、52・・・フラッシュメモリプログラマー、53・・・コピーチップ、54・・・基板、56・・・プログラム記憶装置、57・・・プリンタケーブル、58・・・可搬式記憶媒体、PC・・・パーソナルコンピュータ、WS・・・ワークステーション。

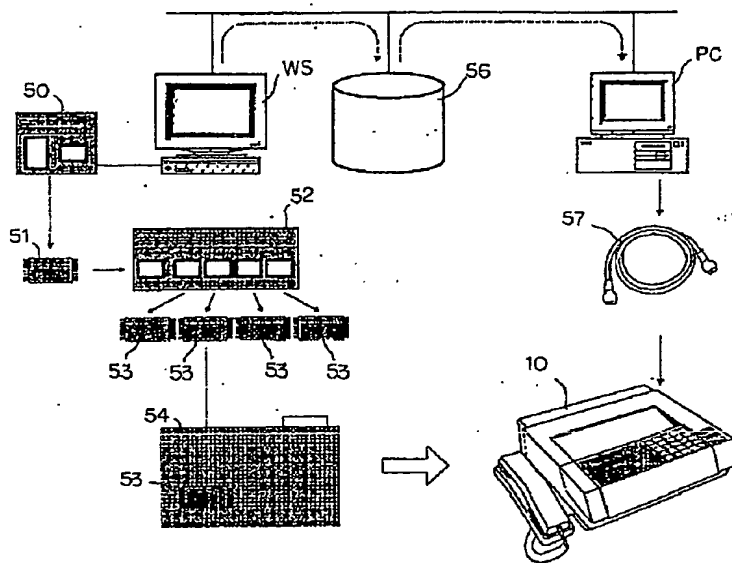
【図1】



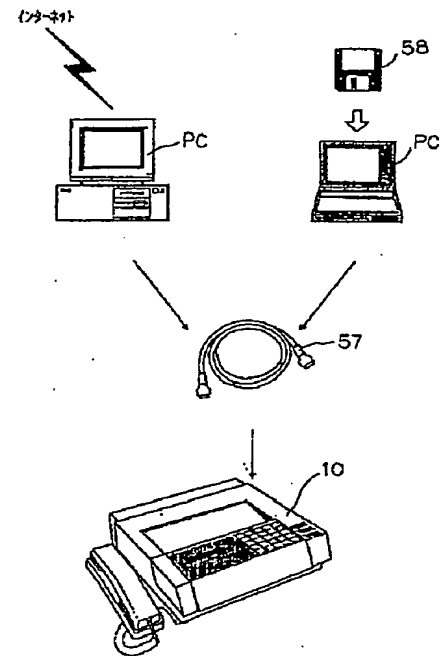
【図 2】



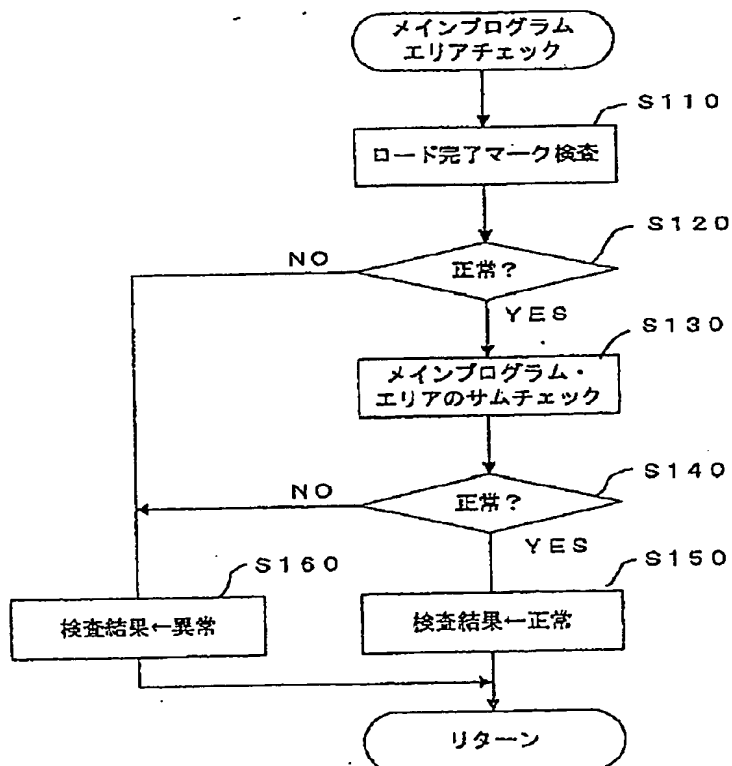
【図 3】



【図 4】



【図 7】



【図 5】

## P J L コマンド

ESC	%	-	1	2	3	4	5	X	@	P	J	L	E	N	T	E	R	=	F	L	A	S	H	CR
-----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----

## 固定文字列

A	B	C		F	A	X		F	L	A	S	H		D	A	T	A
---	---	---	--	---	---	---	--	---	---	---	---	---	--	---	---	---	---

## モデル・国識別用パスワード

A	B	C	9	8	7	6		J	P	N
---	---	---	---	---	---	---	--	---	---	---

## 開始レコード

FFH	00H
-----	-----

## データレコード

属性フィールド	アドレスフィールド	データフィールド	チェックサムフィールド
---------	-----------	----------	-------------

:

データフィールドの内容

プログラム

プログラム
-------

モデル・国識別用

パスワード

A	B	C	9	8	7	6		J	P	N
---	---	---	---	---	---	---	--	---	---	---

ロード完了マーク

P	r	g	r	a	m	O	K
---	---	---	---	---	---	---	---

作成年月日

98H	01H	01H	00H	00H
-----	-----	-----	-----	-----

チェックサム

12H	34H
-----	-----

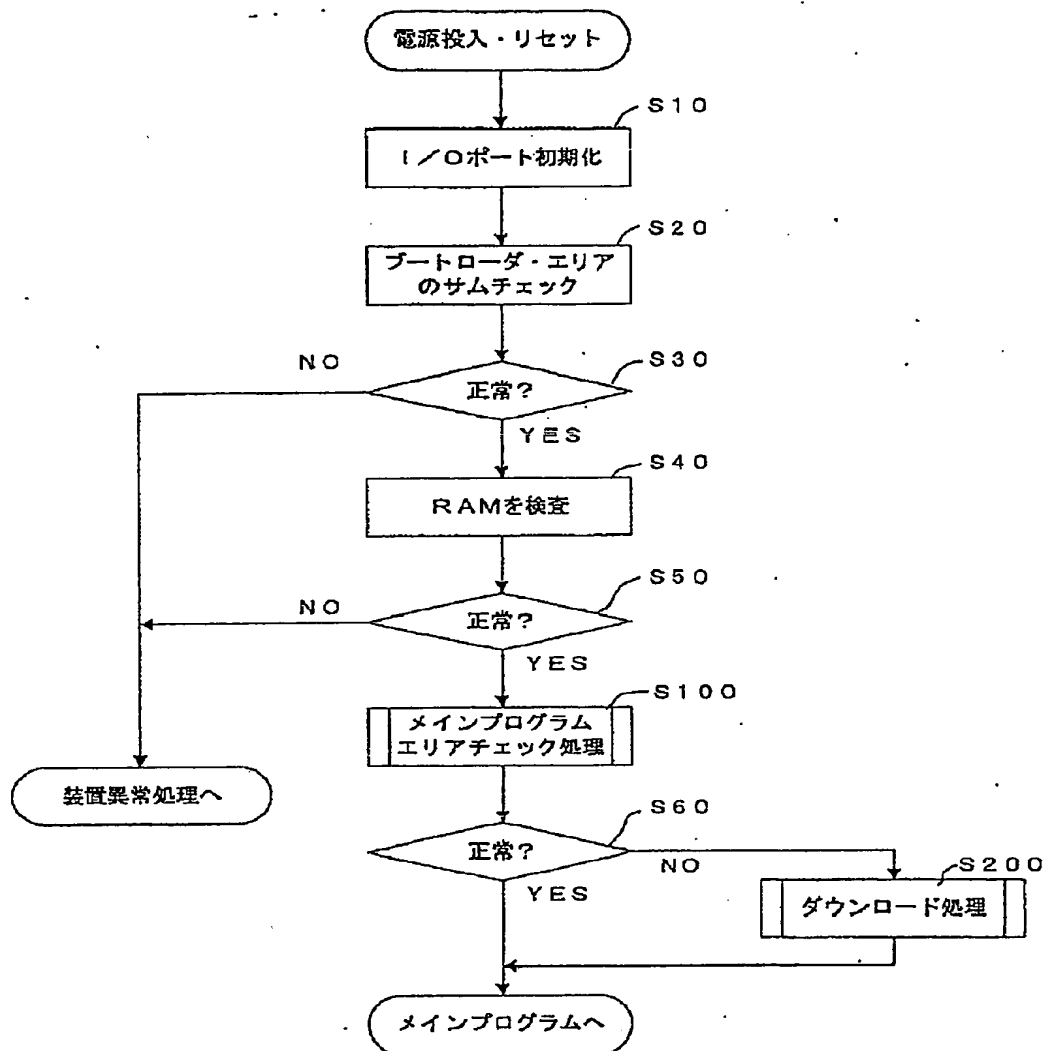
:

属性フィールド	アドレスフィールド	データフィールド	チェックサムフィールド
---------	-----------	----------	-------------

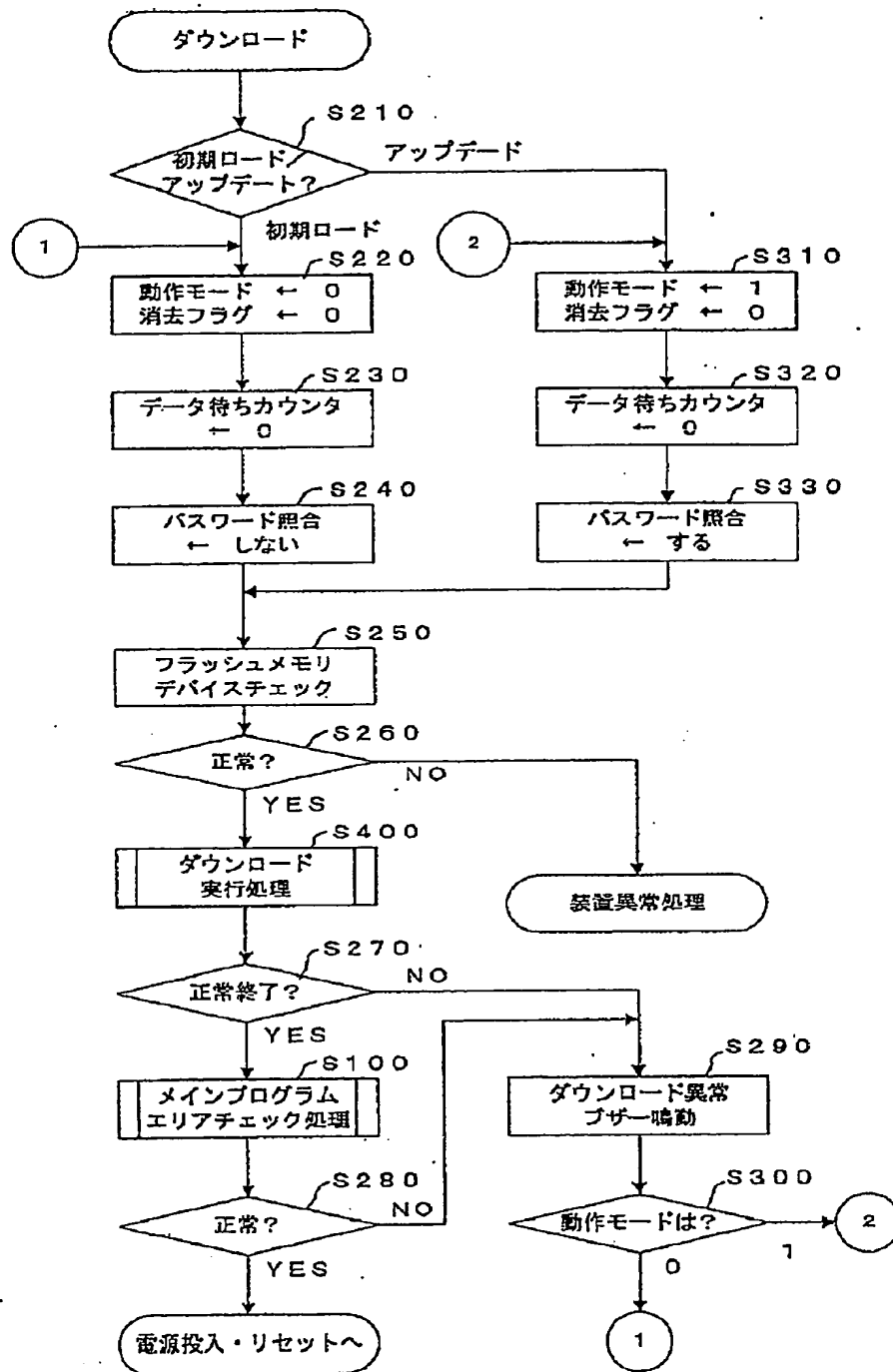
## 最終レコード

00H	00H
-----	-----

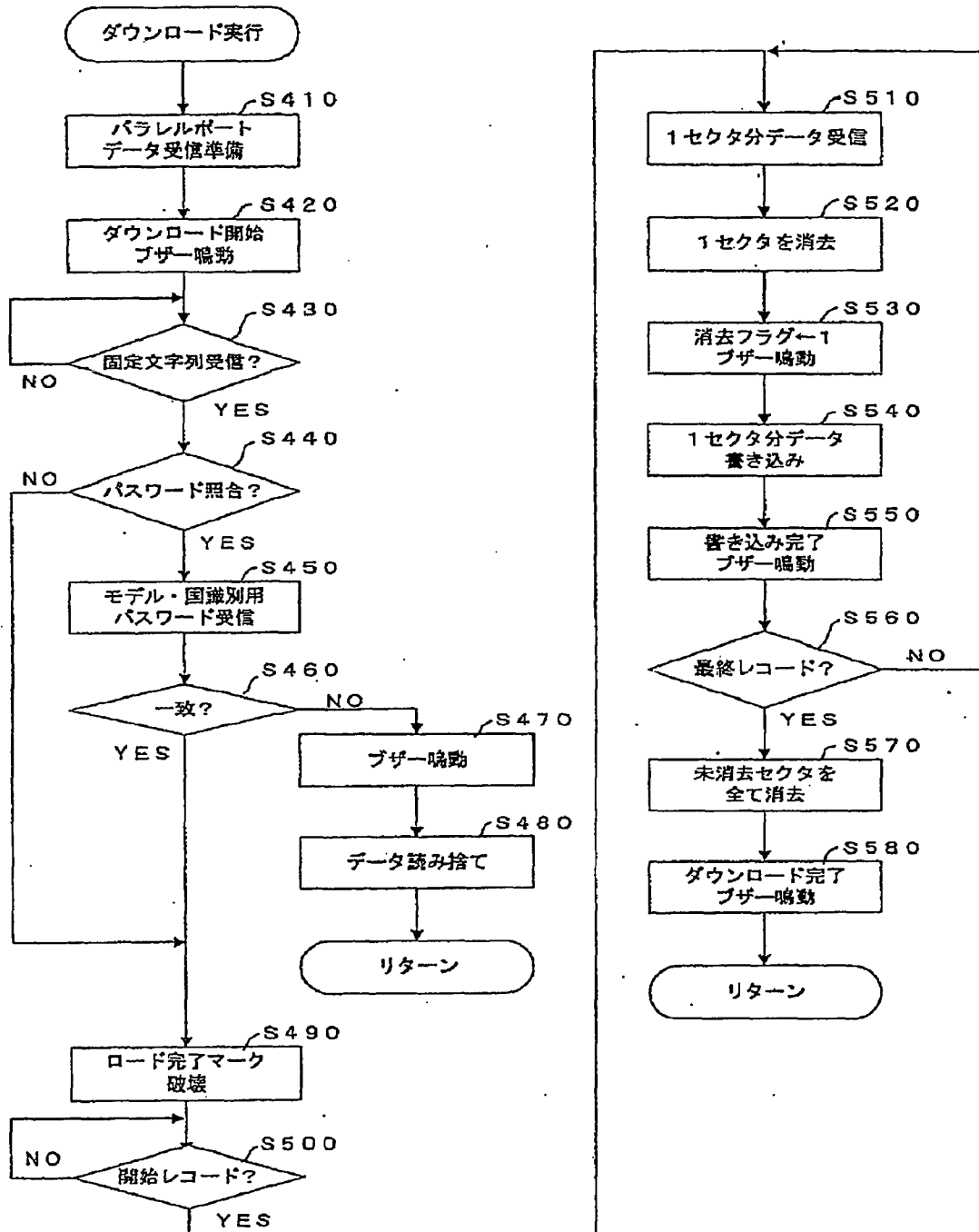
【図 6】



【図 8】



【図 9】



【図 10】

